50 minutes   Closed Book   Nov. 27, 2006
Marks

6   1. Give the expression for the time requirements of the code below. It is sufficient to give the
         O( ) expression for its time. The code calls the functions
                p, an INTEGER function with time requirements $O(k^2)$
                $r(n)$, an INTEGER function with time requirements $O(m)$, m independent of n
         and $s(n)$, an INTEGER function with time requirements $O(k)$, k independent of n

```
local i, total : INTEGER
do
        from
                i := 1
                total := p
        until i > m
        loop
                total := total + r(i)*s(i)
                i := 3*i
        end
end
```
k and m are INTEGER attributes of the class containing this code.

3   2. According to Design by Contract, in the redefinition of a routine in a descendant class, its
         postcondition (i) can only be weakened, (ii) can only be strengthened, (iii) can not be changed,
         or (iv) can be arbitrarily changed.
         (a) Which is correct?
         (b) Explain the rational for this rule.

15   3. Consider the design of two simple classes such that one is a descendant of the other, and both
         have a procedure called p. For one of them, procedure p prints "hi", while for the other it
         prints "bye".   Also, they should be designed so that for the same variable a, the execution of
         two occurrences of the statement $a.p$ in a root class results in different output; i.e., one
         occurrence of $a.p$ results in one version of p being called, and the other occurrence results in
         the other version being called.
         Give the class diagram that shows the features for and relationships amongst the three classes,
         and give the complete Eiffel code for the three classes. The three classes are the two classes
         with procedures called p, and the root class. It is suggested that you keep the classes as simple
         as possible subject to the above constraints. Hint: Use dynamic binding.

6   4. In class, an axiomatic definition was given for the *SIMPLE_LIST* type. In this definition, the functions *make* and *insert_first* were the build operations. Other routines were *is_empty*, *first_item*, and *delete_first*. Give the preconditions and axioms to add to the ADT the following two functions:

         replace_second_item(z), returns a list identical to the original list except that the second item has been replaced by item z.

         sum, returns the sum of the items in the list

         Assume that the items stored in the list have a numeric type so that they can be added.

20   5. Consider a retail store uses a computer system to keep a record of the number of items in inventory for each item type sold. When the inventory gets too low for an item type, more items are ordered from a wholesale supplier. For certain item types, there can be several suppliers for the items, so the system keeps track of the suppliers for such items.

    (a) Give the sequence diagram for the task of adding another supplier. In addition to the information for this supplier, the Universal Product Code (upc) is given for each item type that can be supplied by the supplier. The upc is the identifier stamped on every item to uniquely identify its type. Assume that an object of type ADD_SUPPLIER_COMMAND has already been created. Your sequence diagram should start at the time execution begins in this command object.

    (b) What containers are needed? For each container, what type of data structure should be used to store the items (array, linked list, stack, queue, priority queue, dictionary, keyed dictionary, etc)? Describe why you made this choice.

Total 50   The end